The BagIt Packaging Standard for Interoperability and Preservation

Dr. Raymond Plante

NIST Office of Data and Informatics

National Institute of Standards and Technology

The Data Publication

CfA Dataverse
https://doi.org/10.7910/DVN/28977

HARVARD Dataverse

Search ▾ About User Guide Support Sign Up Log In

Metrics 158 Downloads

Contact Share

Replication data for: Deep 3.8 Micron Observations of the Trapezium Cluster Version 2.0

Muench, August;Alves, Joao;Lada, Charles;Lada, Elizabeth, 2015, "Replication data for: Deep 3.8 Micron Observations of the Trapezium Cluster", https://doi.org/10.7910/DVN/28977, Harvard Dataverse, V2

Cite Dataset ▾

Learn about Data Citation Standards.

Description	This is the data behind the paper, "Deep 3.8 Micron Observations of the Trapezium Cluster," by Lada et al. (2004). It includes FITS image files and reference comparison files that would prove useful for interpreting the FITS image files. A note on the images: two images are given for each position. The "avge" images differ from the "avg" image by a sky offset that was applied for the purposes of creating an image mosaic. Based on observations collected at t he European Southern Observatory, Chile [ESO Program 70.C-0471(A)]. Raw data can be found through the ESO Telescope Bibliography. http://telbib.eso.org/index.php?programid=%2270.C-0471%22, or via direct query (Link). (2015-02-01)
Subject	Astronomy and Astrophysics
Keyword	VLT, ISAAC, L band (3.8 micrometers), Infrared
Related Publication	*Deep 3.8 Micron Observations of the Trapezium Cluster*, Lada et al. 2004, The Astronomical Journal, **128**, 1254 doi: 10.1086/423294

Files | Metadata | Terms | Versions

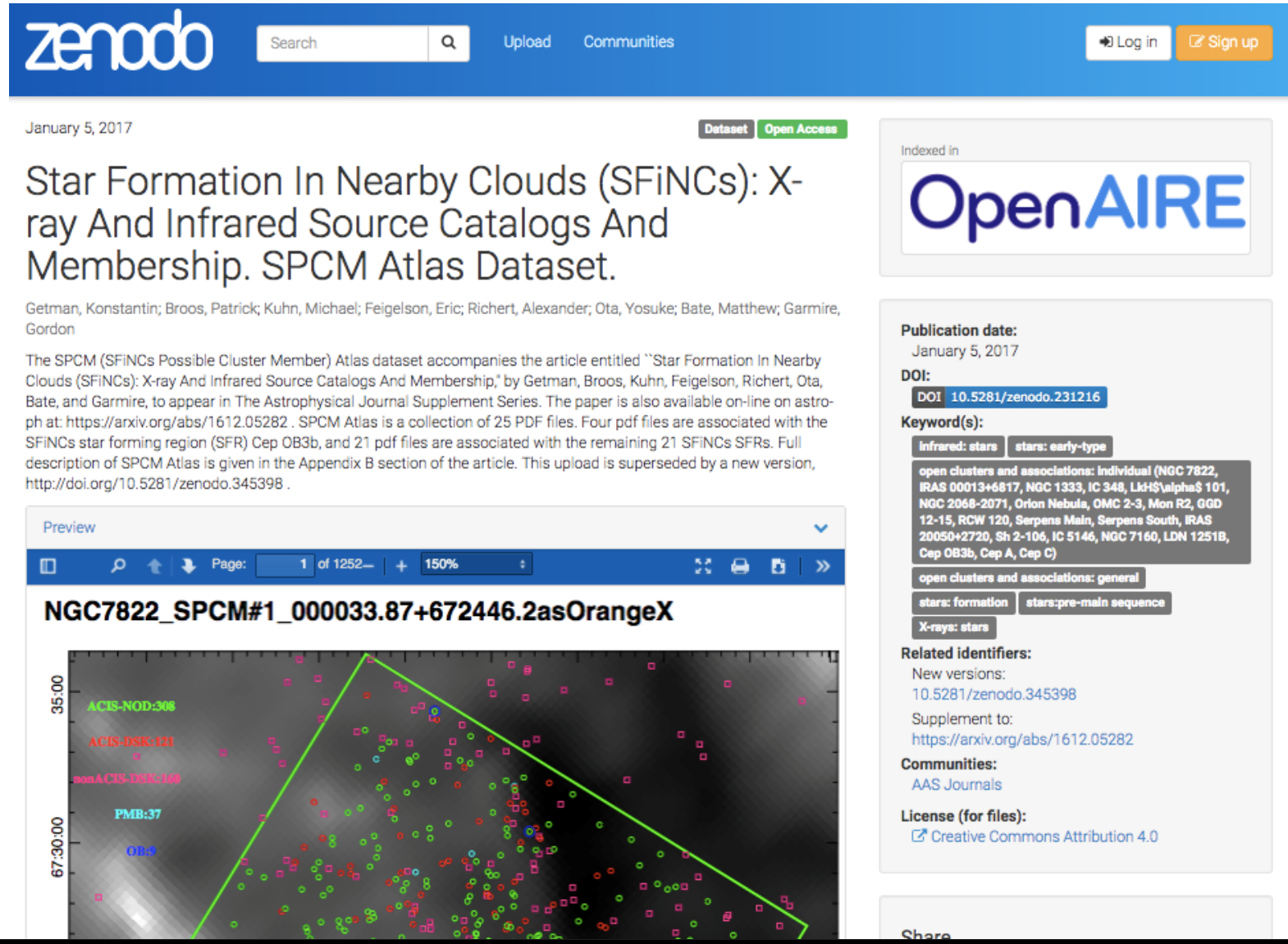
Search this dataset... Find

1 to 10 of 18 Files

The Data Publication

- Title and authorship
- Citation information
- Links to related papers
- Links for downloading data
- Previews and tools

Zenodo Data Archive
https://doi.org/10.5281/zenodo.231216



The NIST Data Publication

Landing page for a NIST Dataset

- Generated automatically from metadata provided by authors
- Modelled as a data publication
- Authors can update their metadata over time to improve presentation and usability
- Updates in underlying data produce a new version
- Support for large and complex datasets
  - File browsing
  - Data cart
  - Globus file transfer

NIST Public Data Repository
https://doi.org/10.18434/M3M956

Data Publication

Experimental test of the intrinsic dimensionality of Hounsfield unit measurements: the CT data

Z. H. Levine, A. R. Peskin, A. Holmgren, E. Garboczi

Contact: Zachary Levine..

Identifier: doi:10.18434/M3M956

Version: 1.1... Last modified: 2018-05-18

Visit Home Page

Description

We present the data supporting "Experimental test of the intrinsic dimensionality of Hounsfield unit measurements" (In preparation). In this study, we passed 34 different substances in separate vials through a computed tomography (CT) scanner at 4 different voltages. At each voltage, we obtained 1824 images (in DICOM format) depicting a sequence of slices through the vials. All 7296 images are provided here. In addition, we provide a table of the substances, their masses, and their positions in the sequence.This dataset deprecates the earlier release of this data (ark:/88434/mds019bfm9). The image and substance table data are exactly the same; however, the image data has been re-arranged to make browsing and downloading more convenient.

Subject Keywords: x-ray computed tomography, medical phantom, Hounsfield unit, volume, shape

Data Access

These data are public.

Files Click on the file/row in the table below to view more details. Total No. files: 475

Name	MediaType	Size	Download
README.txt	text/plain	2.50 kB	
ctBaltimore20170914_02.jpg	image/jpeg	224.2 kB	
ctBaltimoreB20170914.csv	text/csv	1.08 kB	
fig4TheoryExpt.tsv	text/tab-separated-values	6.35 kB	
Compounds			
080kV			
100kV			

Go To ..
Description
Data Access
Record Details
View Metadata
Export JSON
Use
Citation
Fair Use Statement
Find
Similar Resources
Resources by Authors

Scientific Data Preservation Primer

Long-term storage, just in case…

- Error/Disaster recovery
  - Reconstitute individual files or an entire archive
    - if files are corrupted,
    - Major system failures
    - Hacking, …
  - File checksums are important for detecting corruption
- Ensuring Access Long into Future
  - Long-term = decades
  - Can read: survive changes in technology (storage media, OS changes, etc.)
  - Can understand: can read formats, can understand semantics
    - Survive changes in software and people

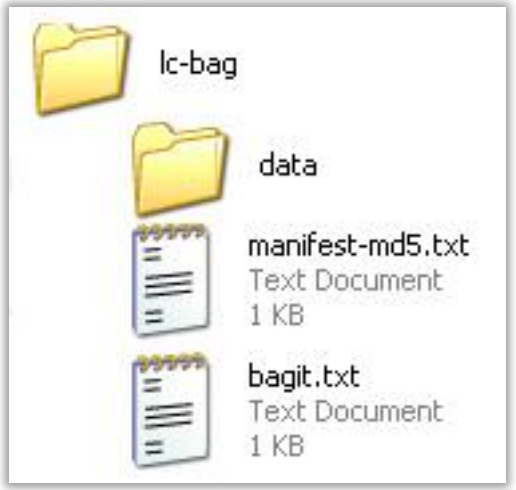
BagIt

Courtesy of the LOC (loc.gov)

Introducing BagIt

- A packaging format for transmitting a digital collection
  - Developed by the California Digital Library, US Library of Congress (LOC), Stanford University
  - A format for packaging data to be ingested into the archive
  - Broader adoption for transferring collections between platforms
- IETF Standard in process
  - Specification (v1.0): https://www.ietf.org/archive/id/draft-kunze-bagit-17.txt
  - LOC maintains libraries for Java, Python on GitHub
    - Includes validator, command-line tool
- Adoption by research repositories as an export format
- At NIST, we are using BagIt as a *preservation format* for data publications

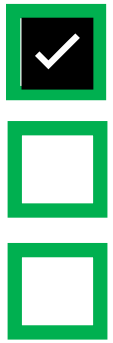
Anatomy of a Bag



Courtesy of the LOC (loc.gov)

- A BagIt "bag" is a directory
  - **`data`** – a payload directory, where the files in the collection live
  - **`bagit.txt`** – BagIt version declaration
  - **`bag-info.txt`** – minimal, machine-readable (but human-oriented) metadata
  - **`manifest-`***`alg`***`.txt`** – list of payload files and checksums
  - *Anything else!*
- The contents of the data directory can preserve the native organization of the collection
- Bag Producers may include additional metadata files
- Directory may serialized in any manner (zip, tar.gz, 7zip)

BagIt Profiles



- Additional rules regarding bag contents
  - Usually specification of additional metadata and/or files
  - Imposed by the producer to support local community features
- bagit-profiles
  - A JSON-formatted description of a BagIt Profile https://github.com/bagit-profiles/bagit-profiles
  - Allows a general validator to test a bag's compliance with a profile. https://github.com/bagit-profiles/bagit-profiles-validator
- It is typically possible for a bag to be compliant with more than one profile

DataONE BagIt Profile

- Require inclusion of an ORE Resource Map
  https://releases.dataone.org/online/api-documentation-v1.2.0/design/DataPackage.html
  - OAI Specification: Object Reuse and Exchange
    http://www.openarchives.org/ore/1.0/toc
  - Used to describe data object aggregations
  - Uses URIs and RDF to express relationships between members of the aggregation
    - Containment, describedBy, describes, …
    - Dublin Core concepts (title, creator, rights, …)
    - Any other ontology concepts
- Not well-defined
  - Deviations from ORE standard

RDA Repository Interoperability WG Profile

- A BagIt Profile to allow greater interoperability between repositories and with data preparation systems https://github.com/RDAResearchDataRepositoryInteropWG/bagit-profiles
- Purpose: to provide minimal metadata for understanding contents
- Details
  - Provide a subdirectory called metadata
  - Include a file called **`metadata/datacite.xml`** that…
    - Conforms to the DataCite Metadata schema
    - Describes the collection
- Status
  - Requires an update to the Data Cite schema to allow use in this context (not require DOI)

“Holey” Bags

- For transmitting large bags
- **`fetch.txt`** – mapping of URLs to file paths within the bag
  - Receiver must download files from URLs to get complete bag
  - Manifest still has checksums that can be checked after retrieval
  - Bags can be metadata-only + **`fetch.txt`**

- Caution when using for preservation….

BagIt for Preservation at NIST

- Preserves the native organization of a data publication as provided by the authors
- Can include full metadata description in local or multiple formats
  - ORE file can describe relationships using community ontologies
- Can include arbitrary ancillary data (figures, previews, …)
- Can serialize and compress for long-term storage
- File checksums of files required
- Can meet Preservation Requirements
  - Can detect corruption via checksums
  - Fully self-contained: no external dependencies
  - Fully self-describing

Disadvantages of Bags for Preservation

- Large collections
  - Not convenient to store very large, serialized bags
    - Makes restoring individual files very inefficient
  - `fetch.txt` file is not a good solution
    - Decades later, can't rely on the existence of an HTTP service

- How do we handle versioning, small updates to publications?
  - Don't want to make 2nd copy of data that has not changed

Proposed Solution: the Multibag BagIt Profile

The Multibag Profile

- Breaks a (large) bag into several smaller bags*
  - *each component bag is a compliant BagIt bag
  - (partially supported by the BagIt Spec. already)
- Define metadata and rules for reconstituting the complete bag
- Allow efficient lookup of location of individual files
- Allow future creation of “errata” bags that contain only changes (new or updated files)

Documented at https://github.com/usnistgov/multibag-py

- **`docs/multibag-profile-spec.md`**
- See also BagIt-profile files

The Multibag Profile: how it works

- A Multibag aggregation is made up of…
  - One “Head” bag
  - Zero or more additional “member” bags
- Head Bag requirements
  - Some additional metadata in the **`bag-info.txt`** file
  - **`multibag/member-bags.tsv`**
    - Lists the names of the bags that make up the aggregation
    - (Accompanying URLs are optional)
    - Order is significant: represents the order that the bags should be unpacked into a common directory to reconstitute the complete bag
  - **`multibag/file-lookup.tsv`**
    - Lists (payload) files that make up the combined bag, mapped to the name of the member bag that each is stored in

The Multibag Profile: creating updates

- An update accomplished by creating…
  - A new Head Bag
  - Zero or more additional member bags
- The new bags contain only those files that have changed
  - "payload" files or metadata files
- In the new Head Bag…
  - it's **`member-bags.tsv`** file can refer to member files that were part of the previous version.
  - its BagIt metadata (**`bag-info.txt`**) includes
    - the version of the revised bag
    - References to the Head Bags of previous versions it deprecates
- Any previous version can be restored by getting the right Head Bag for that version

The Multibage Profile: Future work

- Spec does not yet specify how to deal with an individual file that is very/too large
  - Specify rules for splitting large files across member bags

- Leveraging PIDs
  - Instead of optionally associating a URL with a member bag, associate it with a PID
  - Can't rely on URL still working
  - *May* rely on future PID resolving service

The NIST Preservation Profile

- Complies with the Multibag Profile
  - Many (small) collections will each be covered by a single Head Bag
  - Larger collections will have Head Bags contain metadata only (payload data in member bags).
- NIST-specific metadata
  - **`metadata`** directory with contents that mirrors the hierarchy under **`data`**
  - Contains JSON-LD-formatted metadata for collection and individual files
- ORE file to describe relationships (not implemented yet)
- PREMIS metadata file: records preservation provenance
- **`preservation.log`** – log from the preservation service
- **`ABOUT.txt`** – a human-readable summary of the publication

In Summary

- The BagIt standard has a number features that make it attractive as a preservation format
- Two key disadvantages when using vanilla BagIt:
  - Does not scale well to large bags
  - Does not provide an efficient way to make small changes
- The Multibag BagIt Profile was defined to address these disadvantages
  - I believe it could be more generally useful
  - See https://github.com/usnistgov/multibag-py for specification, ref. software
  - Better integration with PIDs?
- The NIST Preservation Profile builds on Multibag and captures our local metadata